**Name: salini k.b**

**Roll No:33**

**Batch:mca-b**

**Date:24-5-2022**

**Object oriented programming lab**

Co2

Aim

1. Find the area of two rectangles by creating two objects of the Rectangle class.

import java.io.\*;

program

import java.util.\*;

class Rect

{

double width,length,area;

Scanner sc=new Scanner(System.in);

Rect()

{

System.out.print("Enter the length: ");

length=sc.nextDouble();

System.out.print("Enter the width: ");

width=sc.nextDouble();

area=length\*width;

System.out.println("Area of Rectangle: "+area);

}

}

class Demo

{

public static void main(String arg[])

{

System.out.println("First Rectangle ");

Rect r1=new Rect();

System.out.println("Second Rectangle ");

Rect r2=new Rect();

}

}

output



Aim

2]Display the name and marks of 5 students using an array of objects.

program

class Student {

private String name;

private int mark;

private String subject;

public Student(String name, int mark, String subject) {

this.name = name;

this.mark = mark;

this.subject = subject;

}

public String toString() {

return name + " " + mark + " " + subject;

}

}

public class Sample {

public static void main(String[] args) {

Student[] s = new Student[3];

s[0] = new Student("Peter", 31, "maths");

s[1] = new Student("John", 45, "english");

s[2] = new Student("Lisa", 53, "physics");

for (int i = 0; i < 3; ++i) {

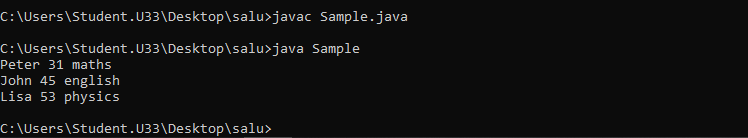
System.out.println(s[i].toString());

}

}

}

output



Aim

3]Create a Java Program to Sort an Array of Objects in Java.

program

import java.util.Arrays;

class Student implements Comparable<Student> {

private String name;

private int age;

private String gender;

public Student (String name, int age, String gender){

this.name = name;

this.age = age;

this.gender = gender;

}

public String toString()

{

return "{" + "name='" + name + '\'' +", gender: "+gender+", "+

"age=" + age + '}';

}

public int getAge() {

return age;

}

public String getGender() {

return gender;

}

public String getName() {

return name;

}

public int compareTo(Student o) {

if (this.age != o.getAge()) {

return this.age - o.getAge();

}

return this.name.compareTo(o.getName());

}

}

public class TestExample {

public static void main(String [] args){

Student[] students = { new Student("achu", 25,"male"), new Student("Salu", 20,"female"),

new Student("san", 22,"male"), new Student("anu", 10,"female") };

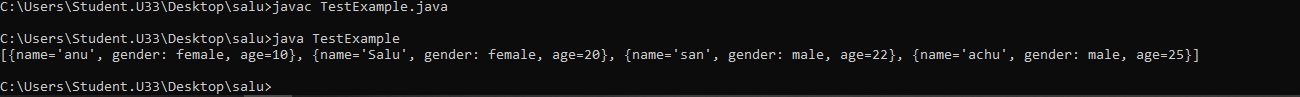
Arrays.sort(students);

System.out.println(Arrays.toString(students));

}

}

output



4. Program to create a class for Employee having attributes eNo, eName eSalary. Read n employ information and Search for an employee given eNo, using the concept of Array of Objects.

5. Create a class for student having attributes RollNo, Name, mark1, mark2, mark3. Read- ‘n’ student information and Find out the total marks of Each student using the concept of Array of Objects

Co3

**Experiment No.: 1**

**Aim**

1. Area of different shapes using overloaded functions

**Procedure**

**c**lass OverloadDemo

{

void area(float x)

{

System.out.println("the area of the square is "+Math.pow(x, 2)+" sq units");

}

void area(float x, float y)

{

System.out.println("the area of the rectangle is "+x\*y+" sq units");

}

void area(double x)

{

double z = 3.14 \* x \* x;

System.out.println("the area of the circle is "+z+" sq units");

}

}

class Overload

{

public static void main(String args[])

{

OverloadDemo obj = new OverloadDemo();

ob.area(8);

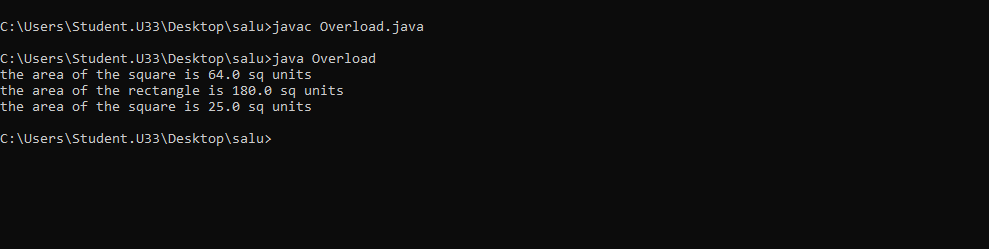
ob.area(12,15);

ob.area(5);

}

}

**Output Screenshot**



**Experiment No:2**

**Aim**

2. Create a class ‘Employee’ with data members Empid, Name, Salary, Address and constructors to initialize the data members. Create another class ‘Teacher’ that inherit the properties of class employee and contain its own data members department, Subjects taught and constructors to initialize these data members and also include display function to display all the data members. Use array of objects to display details of N teachers.

**Procedure**

class EMPS{

public static void main(String[] args) {

Teacher tobj[] = new Teacher[2];

tobj[0] = new Teacher("101","Rekha","Rosevilla",50000,"MCA","DS");

tobj[1] = new Teacher("102","Riya","Deepalayam",110000,"BBA","Commerce");

tobj[0].display();

tobj[1].display();

}

}

class Employees {

String Empid;

String Name;

String Address;

int Salary;

Employees(String id,String name,String addr,int salary){

this.Empid = id;

this.Name = name;

this.Address = addr;

this.Salary = salary;

}

void display(){

System.out.println("EmpID : " + this.Empid);

System.out.println("Name : " + this.Name);

System.out.println("Address : " + this.Address);

System.out.println("Salary : " + this.Salary);

}

}

class Teacher extends Employees{

String Department;

String Subject;

Teacher(String id,String name,String addr,int salary,String dept,String subj){

super(id,name,addr,salary);

this.Department=dept;

this.Subject=subj;

}

void display(){

System.out.println("\*\*\*\*EMPLOYEE DETAILS\*\*\*\*");

super.display();

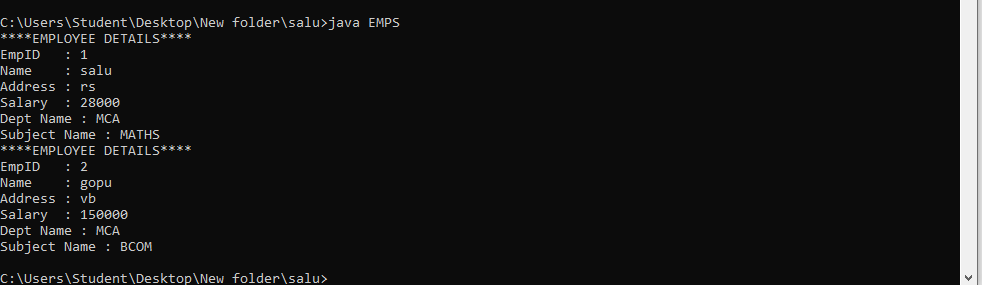
System.out.println("Dept Name : " + this.Department);

System.out.println("Subject Name : " + this.Subject);

}

}

**Output Screenshot**



**Experiment No:3**

**Aim**

3. Create a class ‘Person’ with data members Name, Gender, Address, Age and a constructor to initialize the data members and another class ‘Employee’ that inherits the properties of class Person and also contains its own data members like Empid, Company\_name, Qualification, Salary and its own constructor. Create another class ‘Teacher’ that inherits the properties of class Employee and contains its own data members like Subject, Department, Teacherid and also contain constructors and methods to display the data members. Use array of objects to display details of N teachers.

**Procedure**

import java.util.Scanner;

class Person {

String name, gender, address;

int age;

public Person(String name, String gender, String address, int age) {

this.name = name;

this.gender = gender;

this.address = address;

this.age = age;

}

}

class Employee extends Person {

int empid;

double salary;

String company\_name, qualification;

public Employee(String name, String gender, String address, int age, int empid, String company\_name,

String qualification, double salary) {

super(name, gender, address, age);

this.empid = empid;

this.company\_name = company\_name;

this.qualification = qualification;

this.salary = salary;

}

}

class Teacher extends Employee {

int teacher\_id;

String department, subject;

public Teacher(String name, String gender, String address, int age, int empid, String company\_name,

String qualification, double salary, int teacher\_id, String department, String subject) {

super(name, gender, address, age, empid, company\_name, qualification, salary);

this.teacher\_id = teacher\_id;

this.department = department;

this.subject = subject;

}

void displayDetails(String emp) {

System.out.println("The name of the " + emp + " is: " + this.name);

System.out.println("The gender of the " + emp + " is: " + this.gender);

System.out.println("The address of the " + emp + " is: " + this.address);

System.out.println("The age of the " + emp + " is: " + this.age);

System.out.println("The employee ID of the " + emp + " is: " + this.empid);

System.out.println("The Company name of the " + emp + " is: " + this.company\_name);

System.out.println("The qualification of the " + emp + " is: " + this.qualification);

System.out.println("The salary of the " + emp + " is: " + this.salary);

System.out.println("The teacher ID of the " + emp + " is: " + this.teacher\_id);

System.out.println("The department of the " + emp + " is: " + this.department);

System.out.println("The subject of the " + emp + " is: " + this.subject);

}

}

class arrayMultiLevelInheritance {

public static void main(String[] args) {

int empnum;

Scanner sc = new Scanner(System.in);

System.out.print("Please enter the number of teacher employees you want: ");

empnum = sc.nextInt();

System.out.println("\n");

Teacher[] teachers\_arr = new Teacher[empnum];

for (int i = 0; i < empnum; i++) {

String name, gender, address, company\_name, qualification, department, subject;

int age, empid, teacher\_id;

double salary;

System.out.print("Enter the name of the " + (i + 1) + " teacher : ");

name = sc.next();

System.out.print("Enter the gender of the " + (i + 1) + " teacher : ");

gender = sc.next();

System.out.print("Enter the address of the " + (i + 1) + " teacher : ");

address = sc.next();

System.out.print("Enter the age of the " + (i + 1) + " teacher : ");

age = sc.nextInt();

System.out.print("Enter the emp ID of the " + (i + 1) + " teacher : ");

empid = sc.nextInt();

System.out.print("Enter the company name of the " + (i + 1) + " teacher : ");

company\_name = sc.next();

System.out.print("Enter the qualification of the " + (i + 1) + " teacher : ");

qualification = sc.next();

System.out.print("Enter the salary of the " + (i + 1) + " teacher : ");

salary = sc.nextDouble();

System.out.print("Enter the teacher ID of the " + (i + 1) + " teacher : ");

teacher\_id = sc.nextInt();

System.out.print("Enter the department of the " + (i + 1) + " teacher : ");

department = sc.next();

System.out.print("Enter the subject of the " + (i + 1) + " teacher : ");

subject = sc.next();

teachers\_arr[i] = new Teacher(name, gender, address, age, empid, company\_name,

qualification, salary, teacher\_id, department, subject);

System.out.println("\n");

}

for (int i = 0; i < teachers\_arr.length; i++) {

String txt = (i == 0) ? (i + 1) + "st"

: ((i == 1) ? (i + 1) + "nd" : ((i == 2) ? (i + 1) + "rd" : (i + 1) + "th"));

teachers\_arr[i].displayDetails(txt);

System.out.println("\n");

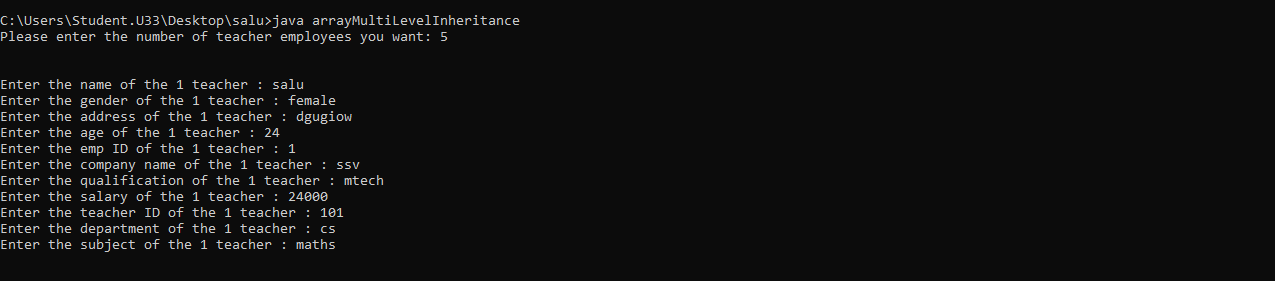
}

sc.close();

}

}

**Output Screenshot**



**Experiment No:4**

**Aim**

4. Write a program has class Publisher, Book, Literature and Fiction. Read the information and print the details of books from either the category, using inheritance.

**Procedure**

import java.util.Scanner;

class Publisher{

int publisher\_id;

String publisher\_name;

Publisher(int publisher\_id, String publisher\_name){

this.publisher\_id= publisher\_id;

this.publisher\_name= publisher\_name;

}

}

class Book extends Publisher{

int book\_id;

String book\_name;

Book(int publisher\_id, String publisher\_name, int book\_id, String book\_name) {

super(publisher\_id, publisher\_name);

this.book\_id= book\_id;

this.book\_name= book\_name;

}

}

class Literature extends Book{

int literature\_id;

String literature\_theme;

Literature(int publisher\_id, String publisher\_name, int book\_id, String book\_name, int literature\_id, String literature\_theme) {

super(publisher\_id, publisher\_name, book\_id, book\_name);

this.literature\_id= literature\_id;

this.literature\_theme= literature\_theme;

}

void displayDetails() {

System.out.println("The publisher ID of the book is: " + this.publisher\_id);

System.out.println("The publisher name of the book is: " + this.publisher\_name);

System.out.println("The Book ID of the book is: " + this.book\_id);

System.out.println("The Book name of the book is: " + this.book\_name);

System.out.println("The Literature ID of the book is: " + this.literature\_id);

System.out.println("The Literature theme of the book is: " + this.literature\_theme);

}

}

class Fiction extends Book{

int fiction\_id;

String fiction\_theme;

Fiction(int publisher\_id, String publisher\_name, int book\_id, String book\_name, int fiction\_id, String fiction\_theme) {

super(publisher\_id, publisher\_name, book\_id, book\_name);

this.fiction\_id= fiction\_id;

this.fiction\_theme= fiction\_theme;

}

void displayDetails() {

System.out.println("The publisher ID of the book is: " + this.publisher\_id);

System.out.println("The publisher name of the book is: " + this.publisher\_name);

System.out.println("The Book ID of the book is: " + this.book\_id);

System.out.println("The Book name of the book is: " + this.book\_name);

System.out.println("The Fiction ID of the book is: " + this.fiction\_id);

System.out.println("The Fiction theme of the book is: " + this.fiction\_theme);

}

}

public class bookInheritance {

public static void main(String[] args) {

Literature literature= new Literature(1000,"jk rowling",200,"harry potter",1958,"Drama");

Fiction fiction= new Fiction(30001, "jk rowling", 3001, "globet of fire", 301, " Fantasy");

literature.displayDetails();

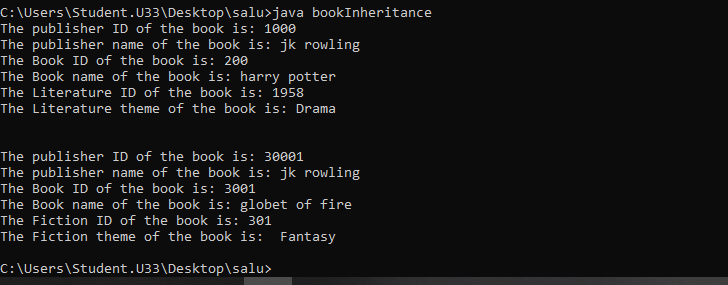
System.out.println("\n");

fiction.displayDetails();

}

}

**Output Screenshot**



**Experiment No:5**

**Aim:**

5. Create classes Student and Sports. Create another class Result inherited from Student and Sports. Display the academic and sports score of a student.

**Procedure**

import java.util.Scanner;

class sports{

String sport;

int Rating;

sports(String spo, int ra){

sport = spo;

Rating = ra;

}

}

class student extends sports{

String Grade;

double Overall\_per;

student(String spo, int ra,String gd, double per ){

super(spo, ra);

Grade = gd;

Overall\_per = per;

}

}

public class result extends student {

result(String spo, int ra,String gd, double per ){

super(spo, ra, gd, per);

}

void display(){

System.out.println("\nSports Details of Student");

System.out.println("Sport :"+sport);

System.out.println("Rating :"+Rating);

System.out.println("\nAcademic Details of Student");

System.out.println("Academic Grade :"+Grade);

System.out.println("Overall percentage :"+Overall\_per);

}

public static void main(String[] args) {

Scanner sc =new Scanner(System.in);

System.out.println("\nEnter the Sports Details of Student");

System.out.println("\n Sport: ");

String a =sc.next();

System.out.println("\n Sport Rating out of 10: ");

int b =sc.nextInt();

System.out.println("\nEnter the Sports Details of Student");

System.out.println("\n Academic Grade: ");

String c =sc.next();

System.out.println("\n Overall percentage: ");

double d =sc.nextDouble();

sc.close();

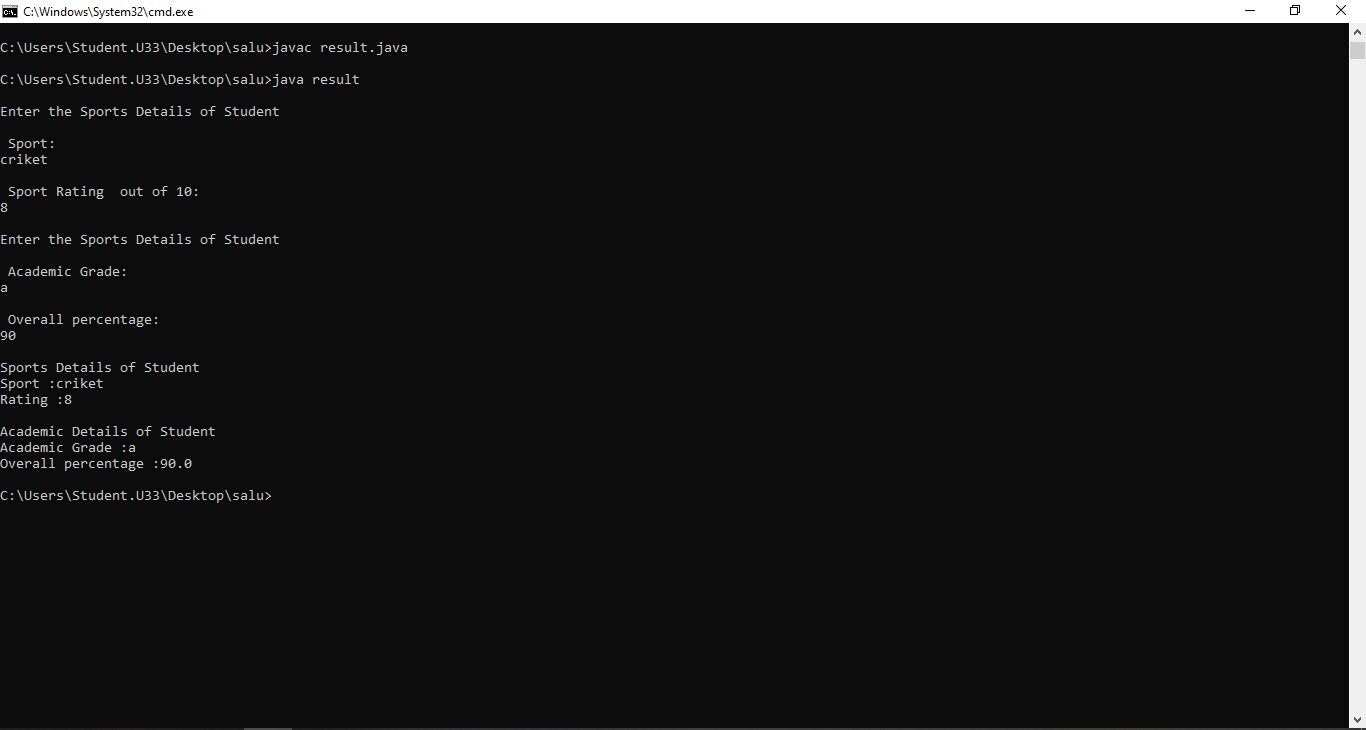
result obj= new result(a,b,c,d);

obj.display();

}

}

**Output Screenshot**



**INTERFACES**

**Aim**

Create an interface having prototypes of functions area and perimeter.create two classes circle and rectangle.create two classes circles and rectangle which implements the above interface.create a menu driven to find area and perimeter of objects.

**program**

import java.util.\*;

interface ShapeCalculate{

double area();

double perimeter();

}

class Circle implements ShapeCalculate{

int radius;

Circle(int radius){

this.radius= radius;

}

public double area() {

return (3.14\*this.radius\*this.radius);

}

public double perimeter() {

return (2\*3.14\*this.radius);

}

}

class Rectangle implements ShapeCalculate{

int length, breadth;

Rectangle(int length, int breadth){

this.length= length;

this.breadth= breadth;

}

public double area() {

return (this.length\*this.breadth);

}

public double perimeter() {

return (2\*(this.length+this.breadth));

}

}

public class shapeInterface {

public static void main(String[] args) {

Scanner sc= new Scanner(System.in);

int choice, isexit=0;

while(true){

System.out.println("Matheatical Operations: \n1. Area of a Rectangle.\n2. Perimeter of a Rectangle.\n3. Area of a circle.\n4. Perimeter of a circle.\n5. Exit\n");

System.out.print("Please enter the choice: ");

choice= sc.nextInt();

switch(choice){

case 1: {

int length, breadth;

System.out.print("\nEnter the length of the rectangle: ");

length= sc.nextInt();

System.out.print("\nEnter the length of the rectangle: ");

breadth= sc.nextInt();

ShapeCalculate rectangleshape= new Rectangle(length, breadth);

System.out.println("The area of the mentioned rectangle is : "+rectangleshape.area()+"sqcm");

break;

}

case 2: {

int length, breadth;

System.out.print("\nEnter the length of the rectangle: ");

length= sc.nextInt();

System.out.print("\nEnter the length of the rectangle: ");

breadth= sc.nextInt();

ShapeCalculate rectangleshape= new Rectangle(length, breadth);

System.out.println("The perimeter of the mentioned rectangle is : "+rectangleshape.perimeter()+"cm");

break;

}

case 3: {

int radius;

System.out.print("\nEnter the radius of the circle: ");

radius= sc.nextInt();

ShapeCalculate circleshape= new Circle(radius);

System.out.println("The area of the mentioned rectangle is : "+circleshape.area()+"sqcm");

break;

}

case 4: {

int radius;

System.out.print("\nEnter the radius of the circle: ");

radius= sc.nextInt();

ShapeCalculate circleshape= new Circle(radius);

System.out.println("The perimeter of the mentioned rectangle is : "+circleshape.perimeter()+"cm");

break;

}

case 5: {

isexit=1;

break;

}

}

if(isexit==1){

break;

}

}

}

}

OUTPUT

